

# Distributed Adaptive Networks: A Graphical Evolutionary Game-Theoretic View

Chunxiao Jiang<sup>\*†</sup>, Yan Chen<sup>\*</sup>, and K. J. Ray Liu<sup>\*</sup>

<sup>\*</sup>Department of Electrical and Computer Engineering, University of Maryland, College Park, MD 20742, USA

<sup>†</sup>Department of Electronic Engineering, Tsinghua University, Beijing, 100084, P. R. China

E-mail: {jcx, yan, kjrlui}@umd.edu

**Abstract**—Distributed adaptive filtering has been considered as an effective approach for data processing and estimation over distributed networks. Most existing distributed adaptive filtering algorithms focus on designing different information diffusion rules, regardless of the nature evolutionary characteristic of a distributed network. In this paper, we study the adaptive network from the game theoretic perspective and formulate the distributed adaptive filtering problem as a graphical evolutionary game. With the proposed formulation, the nodes in the network are regarded as players and the local combination of estimation information from different neighbors is regarded as different strategies selection. We show that this graphical evolutionary game framework is very general and can unify the existing adaptive network algorithms. Based on the framework, we further propose an error-aware adaptive filtering algorithm which does not depend on any network statistical information. Moreover, we use graphical evolutionary game theory to analyze the information diffusion process over the adaptive networks and evolutionarily stable strategy of the system. Finally, simulation results are shown to verify the effectiveness of our analysis and proposed method.

**Index Terms**—Adaptive filtering, graphical evolutionary game, distributed estimation, adaptive networks, data diffusion.

## I. INTRODUCTION

Recently, the concept of adaptive filter network derived from the traditional adaptive filtering was emerging, where a group of nodes cooperatively estimate some parameters of interest from noisy measurements [1]. Such a distributed estimation architecture can be applied to many scenarios, such as wireless sensor networks for environment monitoring, wireless Ad-hoc networks for military event localization, distributed cooperative sensing in cognitive radio networks and so on [2][3]. Compared with the classical centralized architecture, the distributed one is not only more robust when the center node may be dysfunctional, but also more flexible when the nodes are with mobility. Therefore, distributed adaptive filter network has been considered as an effective approach for the implementation of data fusion, diffusion and processing over distributed networks [4].

In a distributed adaptive filter network, at every time instant  $t$ , node  $i$  receives a set of data  $\{d_i(t), \mathbf{u}_i^t\}$  that satisfies a linear regression model as follow

$$d_i(t) = \mathbf{u}_i^t \mathbf{w}^0 + v_i(t), \quad (1)$$

where  $\mathbf{w}^0$  is a deterministic but unknown  $M \times 1$  vector,  $d_i(t)$  is a scalar measurement of some random process  $d_i$ ,  $\mathbf{u}_i^t$  is the  $1 \times M$  regression vector at time  $t$  with zero mean and

covariance matrix  $\mathbf{R}_{u_i} = \mathbb{E}(\mathbf{u}_i^{t*} \mathbf{u}_i^t) > 0$ , and  $v_i(t)$  is the random noise signal at time  $t$  with zero mean and variance  $\sigma_i^2$ . Note that the regression data  $\mathbf{u}_i^t$  and measurement process  $d_i$  are temporally white and spatially independent, respectively and mutually. The objective for each node is to use the data set  $\{d_i(t), \mathbf{u}_i^t\}$  to estimate parameter  $\mathbf{w}^0$ .

In the literatures, many distributed adaptive filtering algorithms have been proposed for the estimation of parameter  $\mathbf{w}^0$ . The incremental algorithms, in which node  $i$  update  $\mathbf{w}$  through combining the observed data sets of itself and node  $i-1$ , were proposed, e.g., incremental LMS algorithm [5]. Unlike the incremental algorithms, the diffusion algorithms allow node  $i$  to combine the data sets from all neighbors, e.g., diffusion LMS [6][7] and diffusion RLS [8]. Besides, the projection-based adaptive filtering algorithms were summarized in [9], e.g., the projected subgradient algorithm [10] and the combine-project-adapt algorithm [11]. In [12], the authors considered the node's mobility and analyzed the mobile adaptive networks.

While achieving promising performance, these traditional distributed adaptive filtering algorithms mainly focused on designing different information combination rules or diffusion rules among the neighborhood. However, on one hand, in a distributed network, to enforce all the nodes to follow some predefined rules may be impractical. Instead, the activities of nodes probably follow some nature evolutionary rules instead of artificially designed rules. On the other hand, although various kinds of combination rules have been developed, there is no general framework that can reveal the unified fundamentals of distributed adaptive filtering problem. In this paper, we will use the evolutionary game theory to formulate the distributed adaptive filtering problem and propose a general framework that can unify the existing algorithms.

The main contributions of this paper are summarized as follows.

- 1) We propose a graphical evolutionary theoretic framework for the distributed adaptive networks, where nodes in the network are regarded as players and the local combination of estimation information from different neighbors is regarded as different strategies selection. Such a framework is very general that can unify existing adaptive filtering algorithms as special cases.
- 2) Based on the proposed game theoretic framework, we further propose an error-aware distributed adaptive filtering algorithm. While achieving better mean-square performances than existing adaptive filtering algorithms, the proposed algorithm does not depend on any network

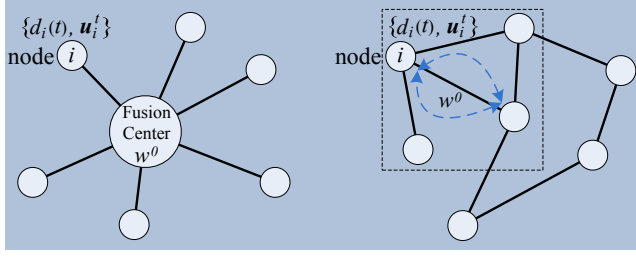


Fig. 1. Left: centralized model. Right: distributed model.

topology and statistical information.

- 3) Using the graphical evolutionary game theory, we analyze the information diffusion process over the adaptive network, and derive the diffusion probability of information from good nodes.
- 4) We prove that the strategy of using information from good nodes is evolutionarily stable strategy either in complete graphs or incomplete graphs.

The rest of this paper is organized as follows. We summarize the existing works in Section II. In Section III, we describe in details how to formulate the distributed adaptive filtering problem as a graphical evolutionary game. We then discuss the information diffusion process over the adaptive network in Section IV, and further analyze the evolutionarily stable strategy in Section V. Simulation results are shown in Section VI. Finally, we draw conclusions in Section VII.

## II. RELATED WORKS

Let us consider an adaptive filter network with  $N$  nodes. If there is a fusion center that can collect information from all nodes, then global (centralized) optimization methods can be used to derive the optimal updating rule for the parameter  $\mathbf{w}$ , where  $\mathbf{w}$  is a deterministic but unknown  $M \times 1$  vector for estimation, as shown in the left part of Fig. 1. For example, in the global LMS algorithm, the parameter updating rule can be written as [6]

$$\mathbf{w}^{t+1} = \mathbf{w}^t + \mu \sum_{i=1}^N \mathbf{u}_i^{t*} (d_i(t) - \mathbf{u}_i^t \mathbf{w}^t), \quad (2)$$

where  $\mu$  is the step size. With (2), we can see that the centralized LMS algorithm requires the information of  $\{d_i(t), \mathbf{u}_i^t\}$  across the whole network, which is generally impractical. Moreover, such a centralized architecture highly relies on the fusion center and will collapse when the fusion center is dysfunctional or some data links are disconnected.

If there is no fusion center in the network, then each node needs to exchange information with the neighbors to update the parameter as shown in the right part of Fig. 1. In the literature, several distributed adaptive filtering algorithms have been introduced, such as distributed incremental algorithms [5], distributed LMS [6][7], and projection-based algorithms [10][11]. These distributed algorithms are based on the classical adaptive filtering algorithms, where the difference is that nodes can use information from neighbors to estimate the parameter  $\mathbf{w}^0$ . Taking one of the distributed LMS algorithms,

TABLE I  
DIFFERENT COMBINATION RULES.

Name	Rule: $A_i(j) =$
Uniform [11][13]	$\frac{1}{n_i}$ , for all $j \in \mathcal{N}_i$
Maximum degree [8][14]	$\begin{cases} \frac{1}{N}, & \text{for } j \neq i, \\ 1 - \frac{n_i-1}{N}, & \text{for } j = i. \end{cases}$
Laplacian [15][16]	$\begin{cases} \frac{1}{n_{\max}}, & \text{for } j \neq i \\ 1 - \frac{n_i-1}{n_{\max}}, & \text{for } j = i. \end{cases}$
Relative degree [8]	$\frac{n_j}{\sum_{k \in \mathcal{N}_i} n_k}$ , for all $j \in \mathcal{N}_i$
Relative degree-variance [6]	$\frac{n_j \sigma_j^{-2}}{\sum_{k \in \mathcal{N}_i} n_k \sigma_k^{-2}}$ , for all $j \in \mathcal{N}_i$

Adaption-then-Combine Diffusion LMS (ATC) [6], as an example, the parameter updating rule for node  $i$  is

$$\begin{cases} \mathbf{x}_i^{t+1} = \mathbf{w}_i^t + \mu_i \sum_{j \in \mathcal{N}_i} c_{i,j} \mathbf{u}_j^{t*} (d_j(t) - \mathbf{u}_j^t \mathbf{w}_j^t), \\ \mathbf{w}_i^{t+1} = \sum_{j \in \mathcal{N}_i} a_{i,j} \mathbf{x}_j^{t+1}, \end{cases} \quad (3)$$

where  $\mathcal{N}_i$  denotes the neighboring nodes set of node  $i$  (including node  $i$  itself),  $c_{i,j}$  and  $a_{i,j}$  are linear weights satisfying the following conditions

$$\begin{cases} c_{i,j} = a_{i,j} = 0, & \text{if } j \notin \mathcal{N}_i, \\ \sum_{j=1}^N c_{i,j} = 1, & \sum_{j=1}^N a_{i,j} = 1. \end{cases} \quad (4)$$

In a practical scenario, since the exchange of full raw data  $\{d_i(t), \mathbf{u}_i^t\}$  among neighbors is costly, the weight  $c_{i,j}$  is usually set as  $c_{i,j} = 0$ , if  $j \neq i$ , as in [6]. In such a case, for node  $i$  with degree  $n_i$  (including node  $i$  itself, i.e., the cardinality of set  $\mathcal{N}_i$ ), we can write the general parameter updating rule as

$$\begin{aligned} \mathbf{w}_i^{t+1} &= A_i(F(\mathbf{w}_1^t), F(\mathbf{w}_2^t), \dots, F(\mathbf{w}_{n_i}^t)), \\ &= \sum_{j \in \mathcal{N}_i} A_i(j) F(\mathbf{w}_j^t), \end{aligned} \quad (5)$$

where  $F(\cdot)$  can be any adaptive filtering algorithm, e.g.  $F(\mathbf{w}_i^t) = \mathbf{w}_i^t + \mu \mathbf{u}_i^{t*} (d_i(t) - \mathbf{u}_i^t \mathbf{w}_i^t)$  for the LMS algorithm,  $A_i(\cdot)$  represents some specific linear combination rule. Eqn. (5) gives a general form of existing distributed adaptive filtering algorithms, where the combination rule  $A_i(\cdot)$  mainly determines the performance. Table I summarizes the existing combination rules, where for all rules  $A_i(j) = 0$ , if  $j \notin \mathcal{N}_i$ .

From Table I, we can see that the weights of the first four combination rules are purely based on the network topology. The disadvantage of such topology-based rules is that, they are sensitive to the spatial variation of signal and noise statistics across the network. The relative degree-variance rule shows better mean-square performance than others, which, however, requires the knowledge of all neighbors' noise variances. As discussed in Section I, all these distributed algorithms are only

focusing on designing the combination rules. Nevertheless, a distributed network is just like a natural ecological system and the nodes are just like individuals in the system, which may spontaneously follow some nature evolutionary rules, instead of some specific artificially predefined rules. Besides, although various kinds of combination rules have been developed, there is no general framework which can reveal the unifying fundamentals of distributed adaptive filtering problems. In the sequel, we will use graphical evolutionary game theory to establish a general framework to unify existing algorithms and give insights of the distributed adaptive filtering problem.

### III. GRAPHICAL EVOLUTIONARY GAME FORMULATION

#### A. Introduction of Graphical Evolutionary Game

Evolutionary game theory (EGT) is originated from the study of ecological biology [17]. It was first adopted to study the gene evolution, where the genes whose strategies are more successful will have higher fitness and higher probability to be reproduced. In such a case, the population fractions of genes whose fitness is higher than the average level of the whole population will tend to grow at a faster rate. Since genes with lower fitness strategies are gradually eliminated during the dynamic process, the stable steady states after the evolution converges must be an Nash equilibrium. Later on, EGT has been widely used to model users' behaviors in image processing [18], as well as communication and networking area [19][20], such as congestion control [21], cooperative sensing [22], cooperative peer-to-peer (P2P) streaming [23] and dynamic spectrum access [24]. In these literatures, evolutionary game has been shown to be an effective approach to model the dynamic social interactions among users in a network. In EGT, there are two important concepts: replicator dynamics and evolutionarily stable strategy [25].

1) *Replicator Dynamics*: EGT differs from the classical game theory by emphasising more on the dynamics and stability of the whole population's strategies, instead of only the property of the equilibrium. In a distributed environment, all players are uncertain about other players actions and utilities. To improve his/her own utility, each player will try different strategies in different rounds of play and learn from the interactions using the methodology of understanding-by-building. During this process, the proportion of players using a certain pure strategy may vary with time. In EGT, replicator dynamics are used to model such a population evolution.

Let us consider an evolutionary game with  $m$  strategies  $\mathcal{X} = \{1, 2, \dots, m\}$ . The payoff matrix,  $U$ , is an  $m \times m$  matrix, whose entries,  $u_{ij}$ , denote the payoff for strategy  $i$  versus strategy  $j$ . The population fraction of strategy  $i$  is given by  $p_i$ , where  $\sum_{i=1}^m p_i = 1$ . The fitness of strategy  $i$  is given by  $f_i = \sum_{j=1}^m p_j u_{ij}$ . For the average fitness of the whole population, we have  $\phi = \sum_{i=1}^m p_i f_i$ . Thus, the replicator dynamic equation is given by [25]

$$\dot{p}_i = \eta p_i (f_i - \phi), \quad (6)$$

where  $\eta$  is a positive scale factor. From (6), we can see that if playing strategy  $i$  can lead to a higher fitness than the average level, the population fraction  $p_i$  will increase and the

increasing rate  $\dot{p}_i/p_i$  is proportional to the difference between  $f_i$  and  $\phi$ . By setting  $\dot{p}_i = 0$  in (6), the theoretical ESS of the game can be calculated through solving the equation. The Wright-Fisher model has been widely adopted to let a group of players converge to the ESS [26], where the strategy updating equation for each player can be written as

$$p_i(t+1) = \frac{p_i(t)f_i(t)}{\phi(t)}. \quad (7)$$

From (7), it can be seen that the strategy updating process in the evolutionary game is similar to the parameter updating process in adaptive filter problem. It is intuitive that we can use evolutionary game to formulate the distributed adaptive filter problem.

2) *Evolutionarily Stable Strategy*: EGT is an effective approach to study how a group of players converges to a stable equilibrium after a period of strategic interactions. Such an equilibrium strategy is defined as the Evolutionarily Stable Strategy (ESS). For an evolutionary game with  $N$  players, a strategy profile  $\mathbf{a}^* = (a_1^*, \dots, a_N^*)$ , where  $a_i \in \mathcal{X}$ , is an ESS if and only if,  $\forall \mathbf{a} \neq \mathbf{a}^*$ ,  $\mathbf{a}^*$  satisfies follows [25]:

$$1) U_i(a_i, \mathbf{a}_{-i}^*) \leq U_i(a_i^*, \mathbf{a}_{-i}^*), \quad (8)$$

$$2) \text{ if } U_i(a_i, \mathbf{a}_{-i}^*) = U_i(a_i^*, \mathbf{a}_{-i}^*), \\ U_i(a_i, \mathbf{a}_{-i}) < U_i(a_i^*, \mathbf{a}_{-i}), \quad (9)$$

where  $U_i$  stands for the utility of player  $i$  and  $\mathbf{a}_{-i}$  denotes the strategies of all players other than player  $i$ . We can see that the first condition is the Nash equilibrium (NE) condition, and the second condition guarantees the stability of the strategy. Moreover, we can also see that a strict NE is always an ESS. If all players adopt the ESS, then no mutant strategy could invade the population under the influence of natural selection. Even if a small part of players may not be rational and take out-of-equilibrium strategies, ESS is still a locally stable state.

3) *Graphical Evolutionary Game*: The classical evolutionary game theory considers a population of  $M$  individuals in a complete graph. However, in many scenarios, players' spatial locations may lead to an incomplete graph structure. Graphical evolutionary game theory is introduced to study the strategies evolution in such a finite structured population [27], where each vertex represents a player and each edge represents the reproductive relationship between valid neighbors, i.e.,  $\theta_{ij}$  denotes the probability that the strategy of node  $i$  will replace that of node  $j$ , as shown in Fig. 2. Graphical EGT focuses on

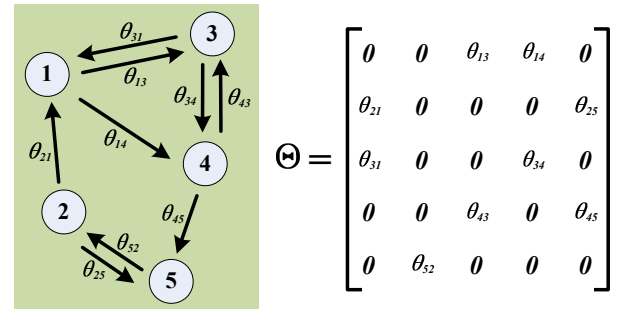


Fig. 2. Graphical evolutionary game model.

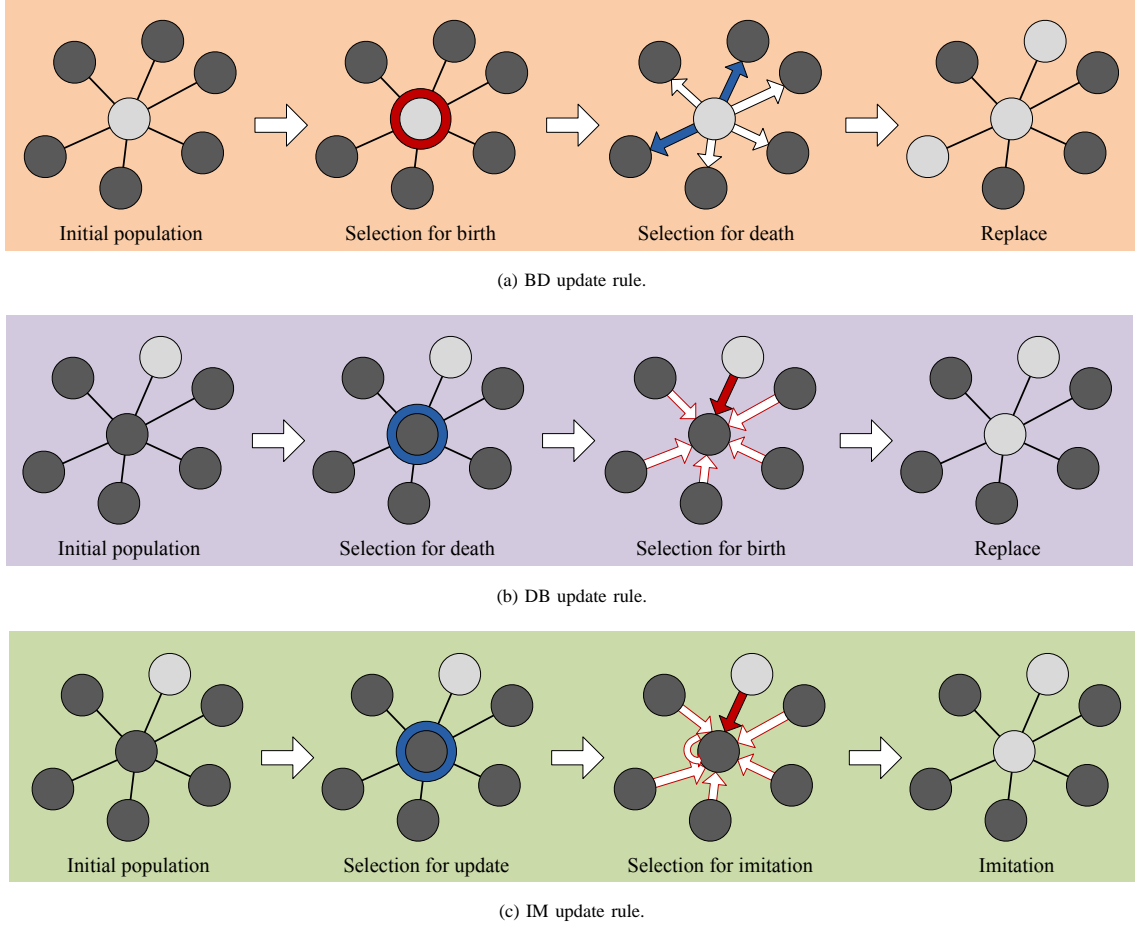


Fig. 3. Three different update rules, where death selections are shown in dark blue and birth selections are shown in red.

analyzing the ability of a mutant gene to overtake a group of finite structured residents. One of the most important research issues in graphical EGT is how to compute the fixation probability, the probability that the mutant will eventually overtake the whole structured population [28]. In this paper, we will use graphical EGT to formulate the dynamic parameter updating process in a distributed adaptive filter network.

### B. Graphical Evolutionary Game Formulation

In graphical EGT, each player updates strategy according to his/her fitness after interacting with neighbors in each round. Similarly, in distributed adaptive filtering, each node updates its parameter  $w$  through incorporating the neighbors' information. In such a case, we can treat the nodes in a distributed filter network as players in a graphical evolutionary game. For node  $i$  with  $n_i$  neighbors, it has  $n_i$  strategies  $\{1, 2, \dots, n_i\}$ , where strategy  $j$  means updating  $w_i^{i+1}$  using the update information from its neighbor  $j$ ,  $A(w_j^t)$ . We can see that (5) represents the adoption of mixed strategy. In such a case, the parameter updating in distributed adaptive filter network can be regarded as the strategy updating in graphical EGT.

We first discuss how players' strategies are updated in graphical EGT, which is then applied to the parameter updating in distributed adaptive filtering. In graphical EGT, the fitness

of a player is locally determined from interactions with all adjacent players, which is defined as [29]

$$f = (1 - \alpha) \cdot B + \alpha \cdot U, \quad (10)$$

where  $B$  is the baseline fitness and  $U$  is the player's payoff which is determined by the predefined payoff matrix. The parameter  $\alpha$  represents the selection intensity, i.e., the relative contribution of the game to fitness. The case  $\alpha \rightarrow 0$  represents the limit of weak selection [30], while  $\alpha = 1$  denotes strong selection, where fitness equals payoff. There are three different strategy updating rules for the evolution dynamics, called as birth-death (BD), death-birth (DB) and imitation (IM) [31].

- BD update rule: a player is chosen for reproduction with the probability being proportional to fitness (Birth process). Then, the chosen player's strategy replaces one neighbor's strategy uniformly (Death process), as shown in Fig. 3-(a).
- DB update rule: a random player is chosen to abandon his/her current strategy (Death process). Then, the chosen player adopts one of his/her neighbors' strategies with the probability being proportional to their fitness (Birth process), as shown in Fig. 3-(b).
- IM update rule: each player either adopts the strategy of one neighbor or remains with his/her current strategy, with the probability being proportional to fitness, as

shown in Fig. 3-(c).

These three kinds of strategy updating rules can be matched to three different kinds of parameter updating algorithms in distributed adaptive filtering. Suppose that there are  $N$  nodes in a structured network, where the degree of node  $i$  is  $n_i$ . We use  $\mathcal{N}$  to denote the set of all nodes and  $\mathcal{N}_i$  to denote the neighborhood set of node  $i$ , including node  $i$  itself.

For the BD update rule, the probability that node  $i$  adopts strategy  $j$ , i.e., using updated information from its neighbor node  $j$ , is

$$P_j = \frac{f_j}{\sum_{k \in \mathcal{N}} f_k} \frac{1}{n_j}, \quad (11)$$

where the first term  $\frac{f_j}{\sum_{k \in \mathcal{N}} f_k}$  is the probability that the neighboring node  $j$  is chosen to reproduction, which is proportional to its fitness  $f_j$ , and the second term  $\frac{1}{n_j}$  is the probability that node  $i$  is chosen for adopting strategy  $j$ . In such a case, the equivalent parameter updating rule for node  $i$  can be written by

$$\mathbf{w}_i^{t+1} = \sum_{j \in \mathcal{N}_i \setminus \{i\}} \left( \frac{f_j}{\sum_{k \in \mathcal{N}} f_k} \frac{1}{n_j} \right) F(\mathbf{w}_j^t) + \left( 1 - \sum_{j \in \mathcal{N}_i \setminus \{i\}} \left( \frac{f_j}{\sum_{k \in \mathcal{N}} f_k} \frac{1}{n_j} \right) \right) F(\mathbf{w}_i^t). \quad (12)$$

Similarly, for the DB updating rule, we can obtain the corresponding parameter updating rule for node  $i$  as

$$\mathbf{w}_i^{t+1} = \frac{1}{N} \sum_{j \in \mathcal{N}_i \setminus \{i\}} \left( \frac{f_j}{\sum_{k \in \mathcal{N}_i} f_k} \right) F(\mathbf{w}_j^t) + \left( 1 - \frac{1}{N} \sum_{j \in \mathcal{N}_i \setminus \{i\}} \left( \frac{f_j}{\sum_{k \in \mathcal{N}_i} f_k} \right) \right) F(\mathbf{w}_i^t). \quad (13)$$

For the IM updating rule, we have

$$\mathbf{w}_i^{t+1} = \sum_{j \in \mathcal{N}_i} \left( \frac{f_j}{\sum_{k \in \mathcal{N}_i} f_k} \right) F(\mathbf{w}_j^t). \quad (14)$$

The performance of adaptive filtering algorithm is usually evaluated by two measures: mean-square deviation (MSD) and excess-mean-square error (EMSE), which are defined as

$$\text{MSD} = \mathbb{E} \|\mathbf{w}^t - \mathbf{w}^0\|^2, \quad (15)$$

$$\text{EMSE} = \mathbb{E} \|\mathbf{u}^t(\mathbf{w}^{t-1} - \mathbf{w}^0)\|^2. \quad (16)$$

Using (12), (13) and (14), we can calculate the network MSD and EMSE of these three update rules according to [6].

### C. Relationship to Existing Distributed Adaptive Filtering Algorithms

In Section II, we have summarized the existing distributed adaptive filtering algorithms in (5) and Table I. In this subsection, we will show that all these algorithms are the special cases of the IM update rule in our proposed graphical EGT framework. Compare (5) with (14), we can see that different fitness definitions are corresponding to different distributed adaptive filtering algorithms in Table I. For the Uniform rule,

TABLE II  
DIFFERENT FITNESS DEFINITION.

Name	Fitness: $f_j =$
Uniform [11][13]	1, for all $j \in \mathcal{N}_i$
Maximum degree [8][14]	$\begin{cases} 1, & \text{for } j \neq i, \\ N - n_i + 1, & \text{for } j = i. \end{cases}$
Laplacian [15][16]	$\begin{cases} 1, & \text{for } j \neq i \\ n_{\max} - n_i + 1, & \text{for } j = i. \end{cases}$
Relative degree [8]	$n_j$ , for all $j \in \mathcal{N}_i$
Relative degree-variance [6]	$n_j \sigma_j^{-2}$ , for all $j \in \mathcal{N}_i$

the fitness can be uniformly defined as  $f_i = 1$  and using the IM update rule, we have

$$\mathbf{w}_i^{t+1} = \sum_{j \in \mathcal{N}_i} \frac{1}{n_i} F(\mathbf{w}_j^t), \quad (17)$$

which is equivalent to the uniform algorithm. Here, the definition of  $f_i = 1$  means the adoption of fixed fitness and weak selection ( $\alpha \ll 1$ ). For the Laplacian rule, when updating the parameter of node  $i$ , the fitness of nodes in  $\mathcal{N}_i$  can be defined as

$$f_j = \begin{cases} 1, & \text{for } j \neq i, \\ n_{\max} - n_i + 1, & \text{for } j = i. \end{cases} \quad (18)$$

From (18), we can see that each node gives more weight to the information from itself through enhancing its own fitness. Similarly, for the Relative-degree-variance rule, the fitness can be defined as

$$f_j = n_j \sigma_j^{-2}, \quad \text{for all } j \in \mathcal{N}_i. \quad (19)$$

Table II summarizes the different fitness definitions corresponding to different combination rules in Table I.

### D. Error-aware Distributed Adaptive Filtering Algorithm

As discussed in Section II, the existing distributed adaptive filtering algorithms either rely on the prior knowledge of network topology or the requirement of additional network statistics. All of them are not robust to a dynamic network, where a node location may change and the noise variance of each node may also vary with time. Considering these problems, we propose an error-aware distributed algorithm based on the intuition that nodes with low mean-square-error (MSE) should be given more weight while nodes with high MSE should be given less weight. The instantaneous MSE of node  $i$ , denoted by  $\beta_i$ , can be calculated by

$$\beta_i^t = \|d_i(t) - \mathbf{u}_i^t \mathbf{w}_i^{t-1}\|^2, \quad (20)$$

where only local data  $\{d_i(t), \mathbf{u}_i^t\}$  are used. We assume that nodes can exchange their instantaneous MSE information with neighbors. The fitness of each node is defined as

$$f_i = e^{-\lambda \beta_i}, \quad (21)$$



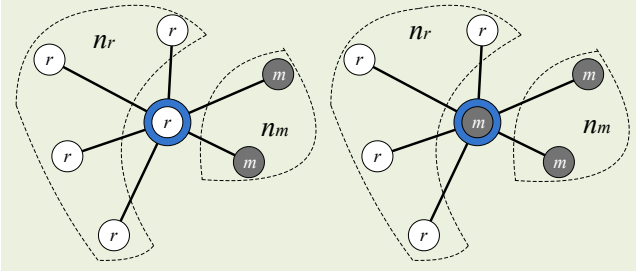


Fig. 4. Graphical evolutionary game model.

where  $\lambda$  is a positive coefficient. In such a case, using the IM update rule, we have

$$\mathbf{w}_i^{t+1} = \sum_{j \in \mathcal{N}_i} \frac{e^{-\lambda \beta_i^t}}{\sum_{k \in \mathcal{N}_i} e^{-\lambda \beta_k^t}} F(\mathbf{w}_j^t). \quad (22)$$

From (22), we can see that the proposed algorithm is not directly dependent with any network topology information. Moreover, it can also adapt to a dynamic environment when the noise variance of some nodes suddenly change since the weights will be immediately adjusted accordingly. In the next section, we will verify the performance of the proposed algorithm through simulation.

#### IV. DIFFUSION ANALYSIS

In a distributed adaptive filter network, there are nodes with good signals, i.e., lower noise variance  $\sigma^2$ , as well as nodes with poor signals. The principal objective of distributed adaptive filtering algorithms is to stimulate the diffusion of good signals to the whole network to enhance the network mean-square performances. In this section, we will use the EGT to analyze such a dynamic diffusion process and derive the close-form expression for the diffusion probability.

In a graphical evolutionary game, the structured population are either residents or mutants. An important concept is the fixation probability, which represents the probability that the mutant will eventually overtake the whole population [32]. Let us consider a local adaptive filter network as shown in Fig. 4, where the hollow points denote common nodes, i.e., nodes with common noise variance  $\sigma_r$ ; and the solid points denote good nodes, i.e., nodes with a lower noise variance  $\sigma_m$ . Here, we adopt the binary signal model to better reveal the diffusion process of good signals. If we regard the common nodes as residents and the good nodes as mutants, the concept of fixation probability in EGT can be applied to analyze the diffusion of good signals in the network. According to the definition of fixation probability, we define the diffusion probability in a distributed filter network as the probability that a good signal can be adopted by all nodes to update parameters in the network.

##### A. Strategies and Payoff Matrix

As shown in Fig. 4, for the node at the center, its neighbors include both common nodes and good nodes. When the center

node updates its parameter  $\mathbf{w}_i$ , it has the following two strategies:

$$\begin{cases} \mathbf{S}_r, & \text{using information from common nodes,} \\ \mathbf{S}_m, & \text{using information from good nodes.} \end{cases} \quad (23)$$

In such a case, we can define the payoff matrix as follow:

$$\begin{matrix} & \mathbf{S}_r & \mathbf{S}_m \\ \mathbf{S}_r & \left( \begin{matrix} \pi^{-1}(\sigma_r, \sigma_r) & \pi^{-1}(\sigma_m, \sigma_r) \end{matrix} \right) \\ \mathbf{S}_m & \left( \begin{matrix} \pi^{-1}(\sigma_r, \sigma_m) & \pi^{-1}(\sigma_m, \sigma_m) \end{matrix} \right) \end{matrix} = \begin{pmatrix} u_1 & u_2 \\ u_3 & u_4 \end{pmatrix}, \quad (24)$$

where  $\pi(x, y)$  represents the EMSE of node with noise variance  $x$  using information from node with noise variance  $y$ . For example,  $\pi(\sigma_r, \sigma_m)$  is the EMSE of node with noise variance  $\sigma_r$  adopting strategy  $\mathbf{S}_m$ , i.e., updating its  $\mathbf{w}$  using information from node with noise variance  $\sigma_m$  which in turn adopts strategy  $\mathbf{S}_r$ . The following Lemma 1 shows the quality of the payoff matrix.

*Lemma 1:* The payoff matrix defined in (24) has the quality as follow

$$u_1 < u_3 < u_2 < u_4. \quad (25)$$

*Proof:* The EMSE  $\pi(x, y)$  is determined by the noise variances of both nodes, as well as the combination rule. According to [33], the optimal  $\pi(x, y)$  can be calculated by

$$\pi(x, y) = c_1 \sigma_1^2 + c_2 \frac{\sigma_x^4}{\sigma_2^2}, \quad (26)$$

$$\begin{cases} c_1 = \frac{\mu \text{Tr}(\mathbf{R}_u)}{4}, & c_2 = \frac{\mu^2 \|\zeta\|^2}{2}, \\ \sigma_1^2 = \frac{2\sigma_x^2 \sigma_y^2}{\sigma_x^2 + \sigma_y^2}, & \sigma_2^2 = \frac{\sigma_x^2 \sigma_y^2}{2}, \end{cases} \quad (27)$$

where  $\zeta = \text{col}\{\zeta_1, \dots, \zeta_N\}$  consists of the eigenvalues of  $\mathbf{R}_u$  (recall that  $\mu$  is the step size and  $\mathbf{R}_u$  is the covariance matrix of the observed regression data  $\mathbf{u}^t$ ). According to (26) and (27), we have

$$\pi(\sigma_r, \sigma_r) = c_1 \sigma_r^2 + 2c_2, \quad (28)$$

$$\pi(\sigma_r, \sigma_m) = c_1 \frac{2\sigma_m^2 \sigma_r^2}{\sigma_m^2 + \sigma_r^2} + 2c_2 \frac{\sigma_r^2}{\sigma_m^2}, \quad (29)$$

$$\pi(\sigma_m, \sigma_r) = c_1 \frac{2\sigma_m^2 \sigma_r^2}{\sigma_m^2 + \sigma_r^2} + 2c_2 \frac{\sigma_m^2}{\sigma_r^2}, \quad (30)$$

$$\pi(\sigma_m, \sigma_m) = c_1 \sigma_m^2 + 2c_2. \quad (31)$$

In such a case, we can see that  $\pi(\sigma_r, \sigma_r) > \pi(\sigma_r, \sigma_m) > \pi(\sigma_m, \sigma_r) > \pi(\sigma_m, \sigma_m)$ , which implies that  $u_1 < u_3 < u_2 < u_4$ . This complete the proof of the lemma. ■

In the following, we will analyze the diffusion process of strategy  $\mathbf{S}_m$ , i.e., the ability of good signals diffusing over the whole network. We consider an adaptive filter network based on a homogenous graph with general degree  $n$  and adopt the IM update rule for the parameter update [34]. Let  $p_r$  and  $p_m$  denote the percentages of nodes using strategies  $\mathbf{S}_r$  and  $\mathbf{S}_m$  in the population, respectively. Let  $p_{rr}$ ,  $p_{rm}$ ,  $p_{mr}$  and  $p_{mm}$  denote the percentages of edge, where  $p_{rm}$  means the percentage of edge on which both nodes use strategy  $\mathbf{S}_r$  and  $\mathbf{S}_m$ . Let  $q_{m|r}$  denote the conditional probability of a node using strategy  $\mathbf{S}_m$  given that the adjacent node is using

strategy  $S_r$ , similar we have  $q_{r|r}$ ,  $q_{r|m}$  and  $q_{m|m}$ . In such a case, we have

$$p_r + p_m = 1, \quad (32)$$

$$q_{r|X} + q_{m|X} = 1, \quad (33)$$

$$p_{XY} = p_Y \cdot q_{X|Y}, \quad (34)$$

$$p_{rm} = p_{mr}, \quad (35)$$

where  $X$  and  $Y$  are either  $r$  or  $m$ . The equations (32-35) imply that the state of the whole network can be described by only two variables,  $p_m$  and  $q_{m|m}$ . In the following, we will calculate the dynamics of  $p_m$  and  $q_{m|m}$  under the IM update rule.

### B. Dynamics of $p_m$ and $q_{m|m}$

According to the IM update rule, a node using strategy  $S_r$  is selected for imitation with probability  $p_r$ . As shown in the left part of Fig. 4, among its  $n$  neighbors (not including itself), there are  $n_r$  nodes using strategy  $S_r$  and  $n_m$  nodes using strategy  $S_m$ , respectively, where  $n_r + n_m = n$ . The percentage of such a configuration is  $\binom{n_m}{n} q_{m|r}^{n_m} q_{r|r}^{n_r}$ . In such a case, the fitness of this node is

$$f_0 = (1 - \alpha) + \alpha(n_r u_1 + n_m u_2), \quad (36)$$

where the baseline fitness is normalized as 1. Among those  $n$  neighbors, the fitness of node using strategy  $S_m$  is

$$f_m = (1 - \alpha) + \alpha([(n-1)q_{r|m} + 1]u_3 + (n-1)q_{m|m}u_4), \quad (37)$$

and the fitness of node using strategy  $S_r$  is

$$f_r = (1 - \alpha) + \alpha([(n-1)q_{r|r} + 1]u_1 + (n-1)q_{m|r}u_2). \quad (38)$$

In such a case, the probability that the node using strategy  $S_r$  is replaced by  $S_m$  is

$$P_{r \rightarrow m} = \frac{n_m f_m}{n_m f_m + n_r f_r + f_0}. \quad (39)$$

Therefore, the percentage of nodes using strategy  $S_m$ ,  $p_m$ , increases by  $1/N$  with probability

$$\text{Prob}\left(\Delta p_m = \frac{1}{N}\right) = p_r \sum_{n_r + n_m = n} \binom{n_m}{n} q_{m|r}^{n_m} q_{r|r}^{n_r} \cdot \frac{n_m f_m}{n_m f_m + n_r f_r + f_0}. \quad (40)$$

Meanwhile, the edges that both nodes use strategy  $S_m$  increase by  $n_m$ , thus, we have

$$\text{Prob}\left(\Delta p_{mm} = \frac{2n_m}{nN}\right) = p_r \binom{n_m}{n} q_{m|r}^{n_m} q_{r|r}^{n_r} \cdot \frac{n_m f_m}{n_m f_m + n_r f_r + f_0}. \quad (41)$$

Similar analysis can be applied to the node using strategy  $S_m$ . According to the IM update rule, a node using strategy  $S_m$  is selected for imitation with probability  $p_m$ . As shown in the right part of Fig. 4, we also assume that there are  $n_r$  nodes using strategy  $S_r$  and  $n_m$  nodes using strategy  $S_m$

among its  $n$  neighbors. The percentage of such a phenomenon is  $\binom{n_m}{n} q_{m|m}^{n_m} q_{r|m}^{n_r}$ . Thus, the fitness of this node is

$$g_0 = (1 - \alpha) + \alpha(n_r u_2 + n_m u_3). \quad (42)$$

Among those  $n$  neighbors, the fitness of node using strategy  $S_m$  is

$$g_m = (1 - \alpha) + \alpha((n-1)q_{r|m}u_3 + [(n-1)q_{m|m} + 1]u_4), \quad (43)$$

and the fitness of node using strategy  $S_r$  is

$$g_r = (1 - \alpha) + \alpha((n-1)q_{r|r}u_1 + [(n-1)q_{m|r} + 1]u_2). \quad (44)$$

In such a case, the probability that the node using strategy  $S_m$  is replaced by  $S_r$  is

$$P_{m \rightarrow r} = \frac{n_r g_r}{n_m g_m + n_r g_r + g_0}. \quad (45)$$

Therefore, the percentage of nodes using strategy  $S_m$ ,  $p_m$ , decreases by  $1/N$  with probability

$$\text{Prob}\left(\Delta p_m = -\frac{1}{N}\right) = p_m \sum_{n_r + n_m = n} \binom{n_m}{n} q_{m|m}^{n_m} q_{r|m}^{n_r} \cdot \frac{n_r g_r}{n_m g_m + n_r g_r + g_0}. \quad (46)$$

Meanwhile, the edges that both nodes use strategy  $S_m$  decrease by  $n_m$ , thus, we have

$$\text{Prob}\left(\Delta p_{mm} = -\frac{2n_m}{nN}\right) = p_m \binom{n_m}{n} q_{m|m}^{n_m} q_{r|m}^{n_r} \cdot \frac{n_r g_r}{n_m g_m + n_r g_r + g_0}. \quad (47)$$

Combining (40) and (46), we have the dynamics of  $p_m$  as

$$\dot{p}_m = \frac{1}{N} \text{Prob}\left(\Delta p_m = \frac{1}{N}\right) - \frac{1}{N} \text{Prob}\left(\Delta p_m = -\frac{1}{N}\right) = \frac{\alpha n(n-1)p_r p_m}{N(n+1)^2} (\gamma_1 u_1 + \gamma_2 u_2 + \gamma_3 u_3 + \gamma_4 u_4) + O(\alpha^2), \quad (48)$$

where the second equality is according to Taylor's Theorem and weak selection assumption with  $\alpha$  goes to zero [35]. In such a case, the payoff obtained from the interactions is considered as limited contribution to the overall fitness of each player. On one hand, the results derived from weak selection often remain as valid approximations for larger selection strength [30]. On the other hand, the weak selection limit has a long tradition in theoretical biology [36]. Moreover, the weak selection assumption can help achieve a close-form analysis of diffusion process and better reveal how the strategy diffuses over the network. The parameters  $\gamma_1$ ,  $\gamma_2$ ,  $\gamma_3$  and  $\gamma_4$  in (48) are given as follows:

$$\gamma_1 = -q_{r|r}[(n-1)(q_{r|r} + q_{m|m}) + 3], \quad (49)$$

$$\gamma_2 = -q_{m|m} - q_{m|r}[(n-1)(q_{r|r} + q_{m|m}) + 2] - \frac{2}{n-1}, \quad (50)$$

$$\gamma_3 = q_{r|r} + q_{r|m}[(n-1)(q_{r|r} + q_{m|m}) + 2] + \frac{2}{n-1}, \quad (51)$$

$$\gamma_4 = q_{m|m}[(n-1)(q_{r|r} + q_{m|m}) + 3]. \quad (52)$$

Similarly, by combining (41) and (47), we have the dynamics of  $p_{mm}$  as

$$\begin{aligned}\dot{p}_{mm} &= \sum_{n_m=0}^n \frac{2n_m}{nN} \text{Prob}\left(\Delta p_{mm} = \frac{2n_m}{nN}\right) \\ &\quad - \sum_{n_m=0}^n \frac{2n_m}{nN} \text{Prob}\left(\Delta p_{mm} = -\frac{2n_m}{nN}\right) \\ &= \frac{2p_{rm}}{(n+1)N} \left(1 + (n-1)(q_{m|r} - q_{m|m})\right) + O(\alpha). \quad (53)\end{aligned}$$

Besides, we can also have the dynamics of  $q_{m|m}$  as

$$\begin{aligned}\dot{q}_{m|m} &= \frac{d}{dt} \left( \frac{p_{mm}}{p_m} \right) \\ &= \frac{2}{(n+1)N} \frac{p_{rm}}{p_m} \left(1 + (n-1)(q_{m|r} - q_{m|m})\right) + O(\alpha). \quad (54)\end{aligned}$$

### C. Diffusion Probability Analysis

The dynamic equation of  $p_m$  in (48) reflects the the dynamic of nodes updating  $w$  using information from good nodes, i.e., the diffusion status of good signals in the network. A positive  $\dot{p}_m$  means that good signals are diffusing over the network, while a negative  $\dot{p}_m$  means that good signals have not been well adopted. The diffusion probability of good signals is closely related to the noise variance of good nodes  $\sigma_m$ . Intuitively, the lower  $\sigma_m$ , the higher probability that good signals can spread the whole network. In this subsection, we will analyze the close-form expression for the diffusion probability.

As discussed at the beginning of Section IV, the state of whole network can be described by only  $p_m$  and  $q_{m|m}$ . In such a case, (48) and (54) can be re-written as functions of  $p_m$  and  $q_{m|m}$

$$\dot{p}_m = \alpha \cdot G_1(p_m, q_{m|m}) + O(\alpha^2), \quad (55)$$

$$\dot{q}_{m|m} = G_2(p_m, q_{m|m}) + O(\alpha). \quad (56)$$

From (55) and (56), we can see that  $q_{m|m}$  converges to equilibrium in a much faster rate than  $p_m$  under the assumption of weak selection. At the steady state of  $q_{m|m}$ , i.e.,  $\dot{q}_{m|m} = 0$ , we have

$$q_{m|m} - q_{m|r} = \frac{1}{n-1}. \quad (57)$$

In such a case, the dynamic network will rapidly converge onto the slow manifold, defined by  $G_2(p_m, q_{m|m}) = 0$ . Therefore, we can assume that (57) holds in the whole convergence process of  $p_m$ . According to (32)-(35) and (57), we have

$$q_{m|m} = p_m + \frac{1}{n-1}(1-p_m), \quad (58)$$

$$q_{m|r} = \frac{n-2}{n-1}p_m, \quad (59)$$

$$q_{r|m} = \frac{n-2}{n-1}(1-p_m), \quad (60)$$

$$q_{r|r} = 1 - \frac{n-2}{n-1}p_m. \quad (61)$$

Therefore, the diffusion process can be characterized by only  $p_m$ . Thus, we can focus on the dynamics of  $p_m$  to derive the diffusion probability, which is given by following *Theorem 1*.

*Theorem 1:* In a distributed adaptive filter network which can be characterized by a  $N$ -node regular graph with degree  $n$ , suppose there are common nodes with noise variance  $\sigma_r$  and good nodes with noise variance  $\sigma_m$ , where each common node has connection edge with only one good node. If each node updates its parameter  $w$  using the IM update rule, the diffusion probability of the good signal is

$$P_{\text{diff}} = \frac{1}{n+1} + \frac{\alpha n N}{6(n+1)^3} (\xi_1 u_1 + \xi_2 u_2 + \xi_3 u_3 + \xi_4 u_4), \quad (62)$$

where the parameters  $\xi_1, \xi_2, \xi_3$  and  $\xi_4$  are as follows:

$$\xi_1 = -2n^2 - 5n + 3, \quad (63)$$

$$\xi_2 = -n^2 - n - 3, \quad (64)$$

$$\xi_3 = 2n^2 + 2n - 3, \quad (65)$$

$$\xi_4 = n^2 + 4n + 3. \quad (66)$$

*Proof:* First, let us define  $m(p_m)$  as the mean of the increment of  $p_m$  per unit time given as follows

$$\begin{aligned}m(p_m) &= \frac{\dot{p}_m}{1/N} \\ &\simeq \frac{\alpha n(n-2)}{(n-1)(n+1)^2} p_m(1-p_m)(ap_m + b). \quad (67)\end{aligned}$$

where the second step is derived by substituting (57)-(61) into (48) and the parameters  $a$  and  $b$  are given as follows:

$$a = (n-2)(n+3)(u_1 - u_2 - u_3 + u_4), \quad (68)$$

$$b = -(n-1)(n+3)u_1 - 3u_2 + (n^2 + n - 3)u_3 + (n+3)u_4. \quad (69)$$

We then define  $v(p_m)$  as the variance of the increment of  $p_m$  per unit time, which can be calculated by

$$v(p_m) = \frac{p_m^2 - (\dot{p}_m)^2}{1/N}, \quad (70)$$

where  $p_m^2$  can be computed by

$$\begin{aligned}p_m^2 &= \frac{1}{N^2} \left( \text{Prob}\left(\Delta p_m = \frac{1}{N}\right) + \text{Prob}\left(\Delta p_m = -\frac{1}{N}\right) \right) \\ &= \frac{2}{N^2} \frac{n(n-2)}{(n-1)(n+1)} p_m(1-p_m) + O(\alpha). \quad (71)\end{aligned}$$

In such a case,  $v(p_m)$  can be approximated by

$$v(p_m) \simeq \frac{2}{N} \frac{n(n-2)}{(n-1)(n+1)} p_m(1-p_m). \quad (72)$$

Suppose the initial percentage of good nodes in the network is  $p_{m0}$ . Let us define  $H(p_{m0})$  as the probability that these good signals can finally be adopted by the whole network, i.e., all nodes can update their own  $w$  using information from good nodes. According to the backward Kolmogorov equation [37],  $H(p_{m0})$  satisfies following differential equation

$$0 = m(p_{m0}) \frac{dH(p_{m0})}{dp_{m0}} + \frac{v(p_{m0})}{2} \frac{d^2 H(p_{m0})}{dp_{m0}^2}. \quad (73)$$

With the weak selection assumption, we can have the approximate solution of  $H(p_{m0})$  as

$$H(p_{m0}) = p_{m0} + \frac{\alpha N}{6(n+1)} p_{m0}(1-p_{m0}) \left( (a+3b) + ap_{m0} \right). \quad (74)$$



Let us consider the worst initial system state that each common node has connection with only one good node, i.e.,  $p_{m0} = \frac{1}{n+1}$ , we have

$$H\left(\frac{1}{n+1}\right) \simeq \frac{1}{n+1} + \frac{\alpha n N}{6(n+1)^3}(a+3b). \quad (75)$$

By substituting (68) and (69) into (75), we can have the close-form expression for the diffusion probability in (62). This completes the proof of the theorem. ■

**Remark:** From (74), we can see that there are two terms constituting the expression of diffusion probability: the initial percentage of strategy  $S_m$ ,  $p_{m0}$  (the initial system state) and the second term representing the changes of system state after beginning, in which  $a+3b$  determines whether  $p_m$  is increasing or decreasing along with the system updating. If  $a+3b < 0$ , i.e., the diffusion probability is even lower than the initial percentage of strategy  $S_m$ , the information from good nodes are shrinking over the network, instead of spreading. Therefore,  $a+3b > 0$  is more favorable for the improvement of the adaptive network performance.

Using *Theorem 1*, we can calculate the diffusion probability of the good signals over the network, which can be used to evaluate the performance of an adaptive filter network. Similarly, the diffusion dynamics and probabilities under BD and DB update rules can also be derived using the same analysis. The following theorem shows an interesting result, which is based on an important theorem in [28], stating that evolutionary dynamics under BD, DB, and IM are equivalent for undirected regular graphs.

*Theorem 2:* In a distributed adaptive filter network which can be characterized by a  $N$ -node regular graph with degree  $n$ , suppose there are common nodes with noise variance  $\sigma_r$  and good nodes with noise variance  $\sigma_m$ , where each common node has connection edge with only one good node. If each node updates its parameter  $w$  using the IM update rule, the diffusion probabilities of good signals under BD and DB update rules are same with that under the IM update rule.

## V. EVOLUTIONARILY STABLE STRATEGY

In the last section, we have analyzed the information diffusion process in an adaptive network under the IM update rule, and derived the diffusion probability of strategy  $S_m$  that using information from good nodes. On the other hand, considering that if the whole network has already chosen to adopt this favorable strategy  $S_m$ , is the current state a stable network state, even though a small fraction of nodes adopt the other strategy  $S_r$ ? In the following, we will answer these questions using the concept of evolutionarily stable strategy (ESS) in evolutionary game theory. As discussed in Section III-A, the ESS ensures that one strategy is resistant against invasion of another strategy [38]. In our system model, it is obvious that  $S_m$ , i.e., using information from good nodes, is the favorable strategy and a desired ESS in the network. In this section, we will check whether strategy  $S_m$  is evolutionarily stable.

### A. ESS in Complete Graphs

We first discuss whether strategy  $S_m$  is an ESS in complete graphs, which is shown by the following theorem.

*Theorem 3:* In a distributed adaptive filter network that can be characterized by complete graphs, strategy  $S_m$  is always an ESS strategy.

*Proof:* In a complete graph, each node meets every other node equally likely. In such a case, according to the payoff matrix in (24), the average payoffs of using strategies  $S_r$  and  $S_m$  are given by

$$U_r = p_r u_1 + p_m u_2, \quad (76)$$

$$U_m = p_r u_3 + p_m u_4, \quad (77)$$

where  $p_r$  and  $p_m$  are the percentages of population using strategies  $S_r$  and  $S_m$ , respectively. Consider the scenario that the majority of the population adopt strategy  $S_m$ , while a small fraction of the population adopt  $S_r$  which is considered as invasion,  $p_r = \epsilon$ . In such a case, according to the definition of ESS in (9), strategy  $S_m$  is evolutionary stable if  $U_m > U_r$  for  $(p_r, p_m) = (\epsilon, 1 - \epsilon)$ , i.e.,

$$\epsilon(u_3 - u_1) + (1 - \epsilon)(u_4 - u_2) > 0. \quad (78)$$

For  $\epsilon \rightarrow 0$ , the left hand side of (78) is positive if and only if

$$“u_4 > u_2” \quad \text{or} \quad “u_4 = u_2 \text{ and } u_3 > u_1”. \quad (79)$$

The (79) gives the sufficient evolutionary stable condition of strategy  $S_m$ . In our system, we have  $u_4 > u_2 > u_3 > u_1$  according to *Lemma 1*, which means that (79) always holds. Therefore, strategy  $S_m$  is always an ESS if the adaptive filter network is a complete graph. ■

### B. ESS in Incomplete Graphs

Let us consider an adaptive filter network which can be characterized by an incomplete regular graph with degree  $n$ . The following theorem shows that strategy  $S_m$  is always an ESS in such an incomplete graph.

*Theorem 4:* In a distributed adaptive filter network which can be characterized by a regular graph with degree  $n$ , strategy  $S_m$  is always an ESS strategy.

*Proof:* Using the pair approximation method [31], the replicator dynamics of strategies  $S_m$  and  $S_r$  on a regular graph of degree  $n$  can be approximated simply by

$$\dot{p}_r = p_r(p_r u'_1 + p_m u'_2 - \phi), \quad (80)$$

$$\dot{p}_m = p_m(p_r u'_3 + p_m u'_4 - \phi), \quad (81)$$

where  $\phi = p_r p_r u'_1 + p_r p_m(u'_2 + u'_3) + p_m p_m u'_4$  is the average payoff, and  $u'_1, u'_2, u'_3$  and  $u'_4$  are given as follows:

$$\begin{cases} u'_1 = u_1, \\ u'_2 = u_2 + u', \\ u'_3 = u_3 - u', \\ u'_4 = u_4. \end{cases} \quad (82)$$

The parameter  $u'$  depends on the three update rules (IM, BD and DB), which is given by [31]

$$\text{IM: } u' = \frac{(n+3)u_1 + u_2 - u_3 - (n+3)u_4}{(n+3)(n-2)}, \quad (83)$$

$$\text{BD: } u' = \frac{(n+1)u_1 + u_2 - u_3 - (n+1)u_4}{(n+1)(n-2)}, \quad (84)$$

$$\text{DB: } u' = \frac{u_1 + u_2 - u_3 - u_4}{n-2}. \quad (85)$$

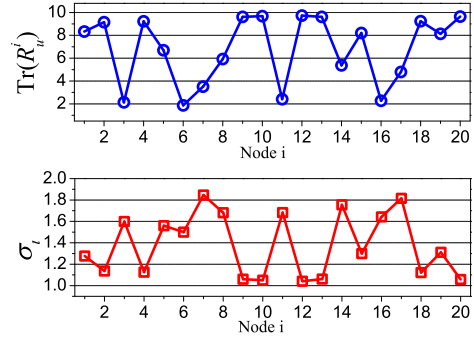
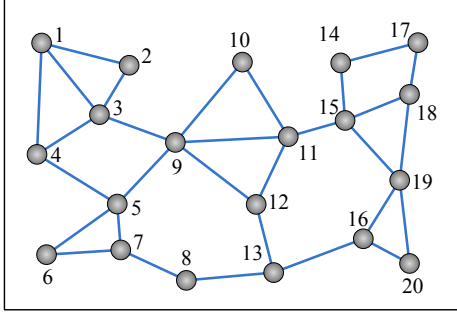


Fig. 5. Network information for simulation, including network topology for 20 nodes (left), trace of regressor covariance  $\text{Tr}(\mathbf{R}_u)$  (right top) and noise variance  $\sigma_i$  (right bottom).

In such a case, the equivalent payoff matrix is

$$\begin{pmatrix} \mathbf{S}_r & \mathbf{S}_m \\ \mathbf{S}_r & \mathbf{S}_m \end{pmatrix} \begin{pmatrix} u_1 & u_2 + u' \\ u_3 - u' & u_4 \end{pmatrix}. \quad (86)$$

According to (79), the evolutionary stable condition for strategy  $\mathbf{S}_m$  is

$$u_4 > u_2 + u'. \quad (87)$$

With *Lemma 1*, we can see that  $u' < 0$  for all three update rules. In such a case, (87) always holds, which means that strategy  $\mathbf{S}_m$  is always an ESS strategy. This completes the proof of the theorem. ■

## VI. SIMULATION RESULTS

In this section, we develop simulations to compare the performances of different adaptive filtering algorithms, as well as to verify the derivation of information diffusion probability and the analysis of ESS.

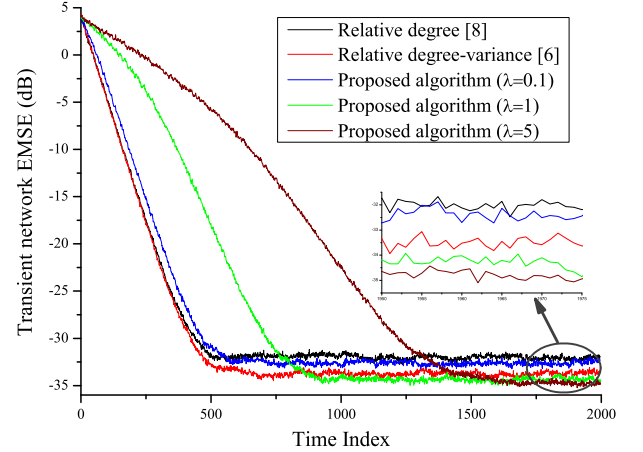
### A. Mean-square Performances Comparison

The network topology used for simulation is shown in the left part of Fig. 5, where 20 randomly nodes are randomly located. The signal and noise power information of each node are also shown in the right part of Fig. 5, respectively. In the simulation, we assume that the regressors with size  $M = 5$ , are zero-mean Gaussian and independent in time and space. The unknown vector is set to be  $\mathbf{w}^0 = \mathbf{1}_5 / \sqrt{2}$  and the step size of the LMS algorithm at each node  $i$  is set as  $\mu_i = 0.01$ . All the simulation results are averaged over 500 independent runnings. All the performance comparisons are conducted among three different kinds of distributed LMS adaptive filtering algorithms as follows:

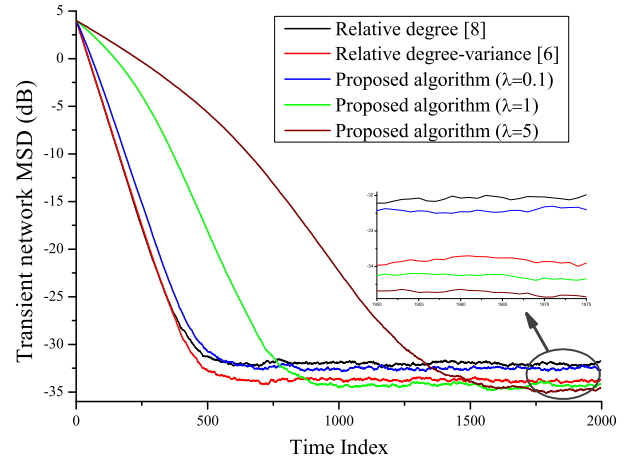
- Relative degree algorithm [8];
- Relative degree-variance algorithm [6];
- Proposed error-aware algorithm with  $\lambda = 0.1, 1$  and 5.

Fig. 6 shows the transient network-performance comparison results among three kinds of algorithms in terms of EMSE and MSD. We can see that our proposed algorithm is always better than the relative degree algorithm with maximal 2.5dB performance enhancement (78% less error). Moreover, the larger  $\lambda$  is set, the lower EMSE and MSD of the proposed

algorithm can be achieved, but with a slower convergence rate. When  $\lambda \geq 1$ , our proposed algorithm can perform better than the relative degree-variance algorithm with about 0.5 – 1dB performance enhancement (12% – 26% less error). As we discussed in Section 2, the relative degree-variance algorithm requires noise variance information of each node, while our proposed algorithm does not.



(a) Network EMSE.



(b) Network MSD.

Fig. 6. Transient performances comparison.

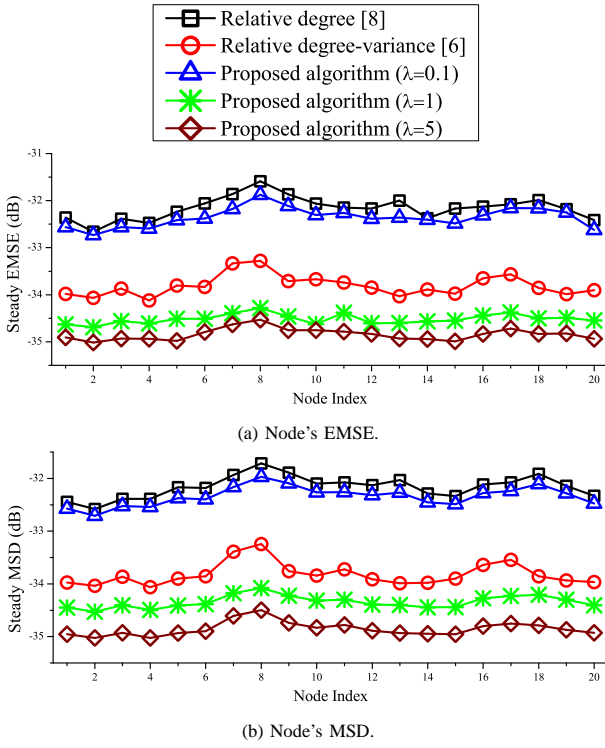


Fig. 7. Steady performances comparison.

Fig. 7 shows the steady-state performances of each node for three kinds of distributed adaptive filtering algorithms in terms of EMSE and MSD. Since the steady-state result is for each node, besides averaging over 500 independent runnings, we average at each node over 100 time slots after the convergence. We can see that the comparison results of steady-state performances are similar to those of the transient performances. Moreover, although there are distinct differences among all nodes' noise variances as shown in Fig. 5-(c), the steady EMSE and MSD of all nodes are similar with each other, which is principally due to the cooperative data sharing and good information diffusion.

To verify the robustness of our proposed algorithm, we also conduct simulations to compare the steady performances of three algorithms when the noise variance of each node is varying over time, as shown in Fig. 8. Based on the noise variances given in Fig. 5-(c), we let the noise variance of each node randomly vary between  $[-50\%, +50\%]$  along with simulation time. We can see that under such circumstances, our proposed algorithm is always better than the relative degree and the relative degree-variance algorithm with about 1 – 3dB performance enhancement (26% – 100% less error). Therefore, the simulation results verify that our proposed error-aware algorithm is resistant to the variation of nodes' noise variances.

### B. Diffusion Probability

In this subsection, we develop simulation to verify the diffusion probability analysis in Section IV. For the simulation setup, three types of regular graphs are generated with degree  $n = 3, 4$  and  $6$ , respectively, as shown in Fig. 9-(a). All these three types of graphs are with  $N = 100$  nodes, where each

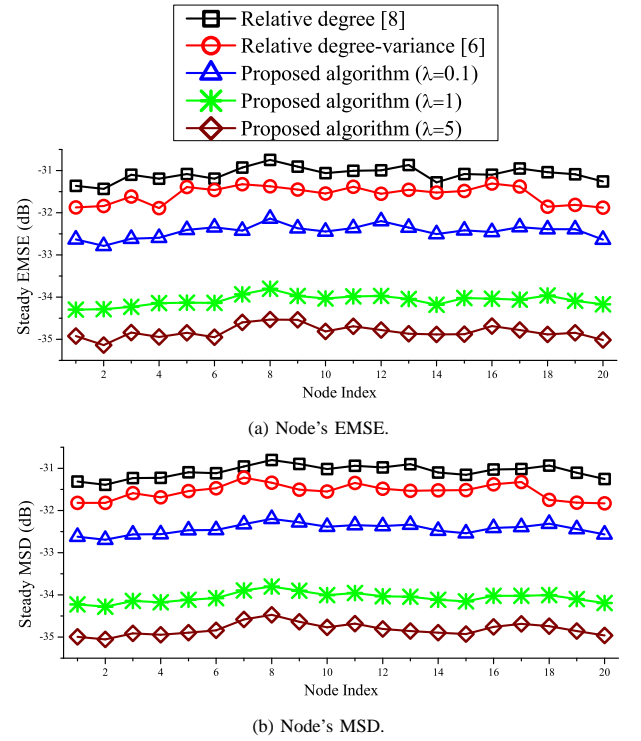
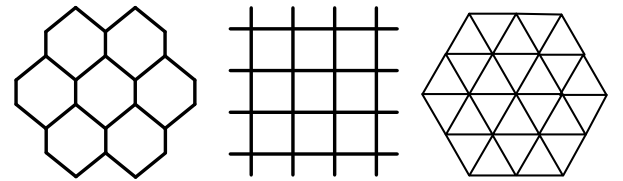
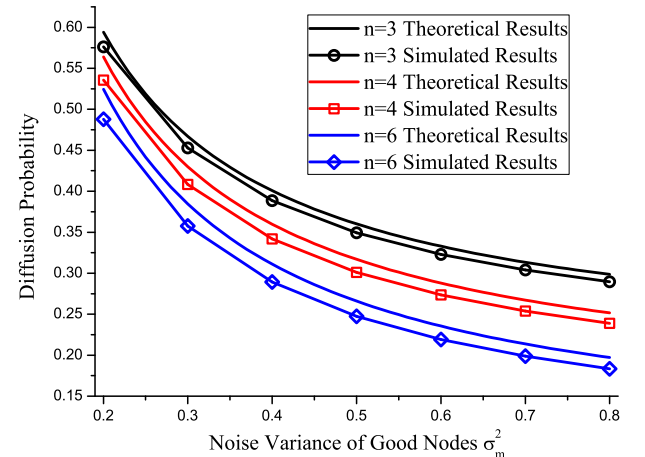


Fig. 8. Steady performances comparison when the noise variances vary over time.

node's trace of regressor covariance is set to be  $\text{Tr}(\mathbf{R}_u) = 10$ , the common nodes's noise variance is set as  $\sigma_r^2 = 1.5$  and the good node's noise variance is set as  $\sigma_m^2 \in [0.2, 0.8]$ . In the simulation, the network is initialized with the state that all

(a) Regular graph structures with degree  $n = 3, 4$  and  $6$ .

(b) Diffusion probability.

Fig. 9. Diffusion probabilities under three types of regular graphs.

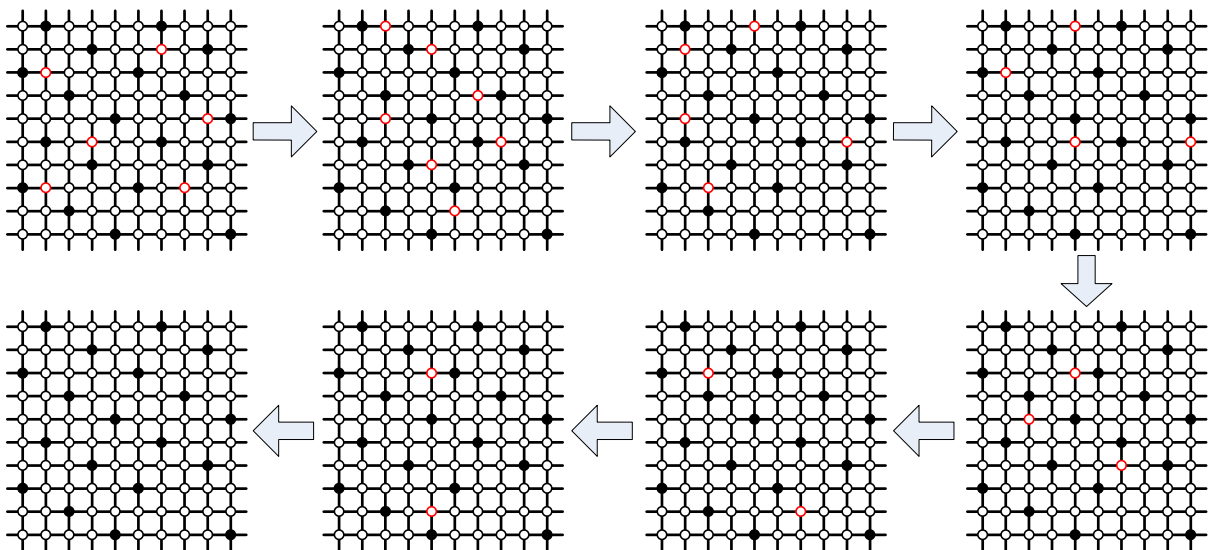


Fig. 10. Strategy updating process in a  $10 \times 10$  grid network with degree  $n = 4$  and number of nodes  $N = 100$ .

common nodes choosing strategy  $S_r$ . Then, at each time step, a randomly chosen node's strategy is updated according to the IM rules under weak selection ( $w = 0.01$ ), as illustrated in Section III-B. The update steps are repeated until either strategy  $S_m$  has reached fixation or the number of steps has reach the limit. The diffusion probability is calculated by the fraction of runs where strategy  $S_m$  reached fixation out of  $10^6$  runs. Fig. 9-(b) shows the simulation results, from which we can see that all the simulated results are basically accord with the corresponding theoretical results and the gaps are due to the approximation during the derivations. Moreover, we can see that the diffusion probability of good signal decreases along with the increase of its noise variance, i.e., better signal has better diffusion capability.

### C. Evolutionarily Stable Strategy

To verify that strategy  $S_m$  is an ESS in the adaptive network, we further simulate the IM update rule on a  $10 \times 10$  grid network with degree  $n = 4$  and number of nodes  $N = 100$ , as shown in Fig. 10 where the hollow points represent common nodes and the solid nodes represent good nodes. In the simulation, all the settings are same with those in the simulation of diffusion probability in Section VI-B, except the initial network setting. The initial network state is set that the majority of nodes adopt strategy  $S_m$  denoted with black color (including both hollow and solid nodes) in Fig. 10, and only a very small percentage of nodes use strategy  $S_r$  denoted with red color. From the strategy updating process of the whole network illustrated in Fig. 10, we can see that the network finally abandons the unfavorable strategy  $S_r$ , which verifies the stability of strategy  $S_m$ .

## VII. CONCLUSION

In this paper, we proposed an evolutionary game theoretic framework to offer a very general view of the distributed adaptive filtering problems and unify existing algorithms.

Based on this framework, we further designed an error-aware adaptive filtering algorithm. The simulation results showed that compared with existing algorithms, our proposed algorithm can achieve better mean-square performances without the knowledge of network statistical information. Using the graphical evolutionary game theory, we further analyzed the information diffusion process in the network under the IM update rule, and proved that the strategy of using information from nodes with good signal is always an ESS. Finally, the final simulation results verified our analysis.

## REFERENCES

- [1] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Distributed processing over adaptive networks," in *Proc. Adaptive Sensor Array Processing Workshop*, Lexington, MA, Jun. 2006, pp. 1–3.
- [2] D. Li, K. D. Wong, Y. H. Hu, and A. M. Sayed, "Detection, classification, and tracking of targets," *IEEE Signal Process. Mag.*, vol. 19, no. 2, pp. 17–29, 2002.
- [3] F. C. R. Jr, M. L. R. de Campos, and S. Werner, "Distributed cooperative spectrum sensing with selective updating," in *Proc. European Signal Processing Conference (EUSIPCO)*, Bucharest, Romania, Aug. 2012, pp. 474–478.
- [4] S. Haykin and K. J. R. Liu, *Handbook on Array Processing and Sensor Networks*. New York: IEEE-Wiley, 2009.
- [5] C. G. Lopes and A. H. Sayed, "Incremental adaptive strategies over distributed networks," *IEEE Trans. Signal Process.*, vol. 55, no. 8, pp. 4064–4077, 2007.
- [6] F. S. Cattivelli and A. H. Sayed, "Diffusion LMS strategies for distributed estimation," *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1035–1048, 2010.
- [7] N. Takahashi, I. Yamada, and A. H. Sayed, "Diffusion least-mean squares with adaptive combiners: Formulation and performance analysis," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4795–4810, 2010.
- [8] F. S. Cattivelli, C. G. Lopes, and A. H. Sayed, "Diffusion recursive least-squares for distributed estimation over adaptive networks," *IEEE Trans. Signal Process.*, vol. 56, no. 5, pp. 1865–1877, 2008.
- [9] S. Theodoridis, K. Slavakis, and I. Yamada, "Adaptive learning in a world of projections," *IEEE Signal Process. Mag.*, vol. 28, no. 1, pp. 97–123, 2011.
- [10] R. L. G. Cavalcante, I. Yamada, and B. Mulgrew, "An adaptive projected subgradient approach to learning in diffusion networks," *IEEE Trans. Signal Process.*, vol. 57, no. 7, pp. 2762–2774, 2009.
- [11] S. Chouvardas, K. Slavakis, and S. Theodoridis, "Adaptive robust distributed learning in diffusion sensor networks," *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4692–4707, 2011.

- [12] S.-Y. Tu and A. H. Sayed, "Mobile adaptive networks," *IEEE J. Sel. Topics Signal Process.*, vol. 5, no. 4, pp. 649–664, 2011.
- [13] V. D. Blondel, J. M. Hendrickx, A. Olshevsky, and J. N. Tsitsiklis, "Convergence in multiagent coordination, consensus, and flocking," in *Proc. Joint 44th IEEE Conf. Decision Control Eur. Control Conf. (CDC-ECC)*, Seville, Spain, Dec. 2005, pp. 2996–3000.
- [14] L. Xiao, S. Boyd, and S. Lall, "A scheme for robust distributed sensor fusion based on average consensus," in *Proc. Information Processing Sensor Networks (IPSN)*, Los Angeles, CA, Apr. 2005, pp. 63–70.
- [15] D. S. Scherber and H. C. Papadopoulos, "Locally constructed algorithms for distributed computations in Ad Hoc networks," in *Proc. Information Processing Sensor Networks (IPSN)*, Berkeley, CA, Apr. 2004, pp. 11–19.
- [16] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Syst. Control Lett.*, vol. 53, no. 1, pp. 65–78, 2004.
- [17] J. M. Smith, *Evolution and the theory of games*. Cambridge, UK: Cambridge University Press, 1982.
- [18] Y. Chen, Y. Gao, and K. J. R. Liu, "An evolutionary game-theoretic approach for image interpolation," in *Proc. IEEE ICASSP*, 2011, pp. 989–992.
- [19] K. J. R. Liu and B. Wang, *Cognitive Radio Networking and Security: A Game Theoretical View*. Cambridge University Press, 2010.
- [20] B. Wang, Y. Wu, and K. J. R. Liu, "Game theory for cognitive radio networks: An overview," *Computer Networks*, vol. 54, no. 14, pp. 2537–2561, 2010.
- [21] E. H. Watanabe, D. Menasché, E. Silva, and R. M. Leão, "Modeling resource sharing dynamics of VoIP users over a WLAN using a game-theoretic approach," in *Proc. IEEE INFOCOM*, 2008, pp. 915–923.
- [22] B. Wang, K. J. R. Liu, and T. C. Clancy, "Evolutionary cooperative spectrum sensing game: how to collaborate?" *IEEE Trans. Commun.*, vol. 58, no. 3, pp. 890–900, 2010.
- [23] Y. Chen, B. Wang, W. S. Lin, Y. Wu, and K. J. R. Liu, "Cooperative peer-to-peer streaming: an evolutionary game-theoretic approach," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 20, no. 10, pp. 1346–1357, 2010.
- [24] C. Jiang, Y. Chen, Y. Gao, and K. J. R. Liu, "Joint spectrum sensing and access evolutionary game in cognitive radio networks," *submitted to IEEE Trans. Wireless Commun.*, 2013.
- [25] R. Cressman, *Evolutionary Dynamics and Extensive Form Games*. Cambridge, MA: MIT Press, 2003.
- [26] R. Fisher, *The Genetical Theory of Natural Selection*. Clarendon Press, 1930.
- [27] E. Lieberman, C. Hauert, and M. A. Nowak, "Evolutionary dynamics on graphs," *Nature*, vol. 433, pp. 312–316, 2005.
- [28] P. Shakarian, P. Roos, and A. Johnson, "A review of evolutionary graph theory with applications to game theory," *Biosystems*, vol. 107, no. 2, pp. 66–80, 2012.
- [29] M. A. Nowak and K. Sigmund, "Evolutionary dynamics of biological games," *Science*, vol. 303, pp. 793–799, 2004.
- [30] H. Ohtsuki, M. A. Nowak, and J. M. Pacheco, "Breaking the symmetry between interaction and replacement in evolutionary dynamics on graphs," *Phys. Rev. Lett.*, vol. 98, no. 10, p. 108106, 2007.
- [31] H. Ohtsuki and M. A. Nowak, "The replicator equation on graphs," *J. Theor. Biol.*, vol. 243, pp. 86–97, 2006.
- [32] M. Slatkin, "Fixation probabilities and fixation times in a subdivided population," *J. Theor. Biol.*, vol. 35, no. 3, pp. 477–488, 1981.
- [33] X. Zhao and A. H. Sayed, "Performance limits for distributed estimation over LMS adaptive networks," *IEEE Trans. Signal Process.*, vol. 60, no. 10, pp. 5107–5124, 2012.
- [34] H. Ohtsuki, C. Hauert, E. Lieberman, and M. A. Nowak, "A simple rule for the evolution of cooperation on graphs and social networks," *Nature*, vol. 441, pp. 502–505, 2006.
- [35] F. Fu, L. Wang, M. A. Nowak, and C. Hauert, "Evolutionary dynamics on graphs: Efficient method for weak selection," *Phys. Rev.*, vol. 79, no. 4, p. 046707, 2009.
- [36] G. Wild and A. Traulsen, "The different limits of weak selection and the evolutionary dynamics of finite populations," *J. Theor. Biol.*, vol. 247, no. 2, pp. 382–390, 2007.
- [37] W. J. Ewens, *Mathematical population genetics: theoretical introduction*. Springer, New York, 2004.
- [38] H. Ohtsuki and M. A. Nowak, "Evolutionary stability on graphs," *J. Theor. Biol.*, vol. 251, no. 4, pp. 698–707, 2008.